MULTIMEDIA UNIVERSITY

STUDENT ID NO | | | | | | | | | | |

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 3, 2018/2019

DCS5068 – DATA STRUCTURE AND ALGORITHMS

(For DIT students only)

31 MAY 2019
9.00 a.m – 11.00 a.m
(2 Hours)

INSTRUCTIONS TO STUDENTS

1. This examination paper consists of **SEVEN (7)** pages excluding cover page.
2. There are **FOUR (4)** structured questions.
3. Write all your answers in the Answer Booklet provided.

Instruction: Write your answers in the answer booklet provided. Total marks are 50.

QUESTION 1 [10 Marks]

a) The program in Figure 1 is written to get two (2) petrol variants from the user together with the prices. The program will then display the two (2) variants and their prices. Complete the following program to get the intended output as described. (5 marks)

```
#include<iostream>
#include<iomanip>
using namespace std;

__________(i)__________

__________(ii)_________

int main()
{
   __________ (iii) __________

   for(int i=0;i<2;i++)
   {
      cout<<"Enter variant "<<i+1<<" : ";
      _____________(iv)_____________

      cout<<"Enter price per liter : ";
      _____________(v)_____________
   }

   __________ (vi) __________

   return 0;
}

__________ (vii) _____________
{
     cout<<"Variant"<<setw(10)<<"Price\n"<<endl;
     for(int i=0;i<2;i++)
     {
      _______ (viii) __________

     }
}
```

Figure 1

Continued ...

i. Create a structure named `petrol` with two (2) data members :
   - `variant` to store petrol variant's name
   - `price` to store the petrol price per liter

ii. Write a void function prototype for function named `display()` that will accept an array type structure.

iii. Declare an instance of the structure named `p` as an array with size two (2).

iv. Store variant's name input from the user into data member `variant`.

v. Store price input from the user into data member `price`.

vi. Call function `display( )` and send array `p` as argument.

vii. Write function header for function named `display()` that will accept an array type structure.

viii. Display the variants' names and prices (set the prices to two (2) decimal point value).

Continued ...

b) Write the output for the program in Figure 2. (5 marks)

```
#include<iostream>
using namespace std;

void add(int &v1,int v2);

int main()
{
      int array[]={10,20,30,40};
      int *x, *y;
      int a=2, b=4;

      x = array;
      b = *(x+3) + (*x+2);
      cout<<"b = "<<b<<endl;
      y = &b;
      *x = *y + 4;
      cout<<array[0]<<endl;
      int &c = a;
      c = x[1] + *y;
      cout<<"a = "<<a<<endl;
      add(c,array[2]);
      cout<<"c = "<<c<<endl;
      cout<<*(x+2)<<endl;

      return 0;
}

void add(int &v1,int v2)
{
      v1 = v1 + 2;
      v2 = v2 + 2;
}
```

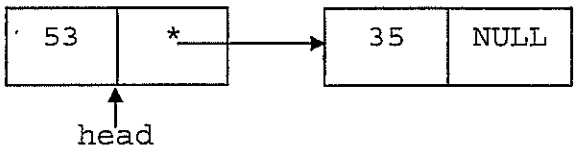
Figure 2

Continued ...

QUESTION 2 [10 Marks]

a) Answer the following questions on linked list operation.

i. Show and draw the trace diagram for the C++ code in Figure 3. Assume that the node consists of two members, `data` (type int) and `link`.
(`link`, `head`, `temp` and `ptr` are pointers of the type node). (3 marks)




```
struct node{
   int data;
   node *link;
}*head, *ptr, *temp;

i.      ptr = new node;
        ptr -> data = 77;

ii.     ptr -> link = head -> link;

iii.    head -> link = ptr;

iv.     temp = new node;
        temp -> data = 105;

v.      temp -> link = ptr -> link;
        head -> link = temp;
        ptr -> link = NULL;
        delete ptr;
        temp = head;
```

Figure 3

ii. Write code segment to print the element in the linked list that you have drawn in Question 2(a)(i). (3 marks)

Continued ...

b) Complete the pop () function in Figure 4 to remove an element from a linked stack. (4 marks)

```
struct node
{
     int data;
     node *next;
};

node *temp, *top;

int empty()
{
    if(top==NULL)
      return 1;
    else
      return 0;
}

int pop()
{
     int num;
     node *temp;
     if(!empty())
     {
          _____(i)_____;

          _____(ii)_____;

          _____(iii)_____;

          _____(iv)_____;

          return num;
     }
     else
     {
          cout<<"Stack is empty\n";
          return 0;
     }
}
```

Figure 4

Continued ...

QUESTION 3 [15 Marks]

a) Given an array of eight (8) elements in Figure 5, find value 40 in the array using binary search algorithms. State clearly the value of `first`, `middle` and `last` index in each steps. (5 marks)

80	75	70	50	40	35	20	5

Figure 5

b) The program in Figure 6 is written to find a target value in an array using linear search algorithms. Complete the code at part marked X. Use `while` loop to traverse through the element one by one. The loop should stop looping either when it has reach the end of the array or when the target value has been found. Use `if` statement in the loop to compare the element in the array with the target value. (4 marks)

```
#include<iostream>
using namespace std;

int main()
{
      int target, index, i = 0;
      bool found = 0;
      int array[] = {10,20,30,40,50};
      cout<<"Enter value to search : ";
      cin>>target;

      X

      if(found==1)
        cout<<"Target found at index "<<index<<endl;
      else
        cout<<"Target not found."<<endl;

      return 0;
}
```

Figure 6

Continued ...

c) Write function definition for a function named `truncate( )` that will received an integer value of nine (9) digits and return an integer value after the truncation. The function should truncate the first two (2) digits and the last three (3) digits of any given nine (9) digits integer. (3 marks)

d) The following integers are to be stored in a hash table of size eight (8). Calculate the index for each integers using hash function, `hash (key) = key mod 10`. If collision occur, resolve using linear probing. (3 marks)

106, 203, 104, 206, 307, 406

QUESTION 4 [15 Marks]

a) Given an array with eight (8) elements in Figure 7, use **Insertion Sort to** sort the following numbers in **ascending order.** Draw the sequence of diagram to depict the process. (4 marks)

145	43	82	13	107	33	62	7	9

Figure 7

b) Draw a binary search tree using the following lists of keys inserted in the given order:

million, reason, give, private, support, care, window, home, unit (5 marks)

c) From the drawn binary search tree in Question 4(b) :

i. What is the height of the drawn binary search tree? (1 mark)

ii. List the descendants of node 'give'. (1 mark)

iii. Write the Breadth First Search (BFS). (2 marks)

iv. List the leaf nodes. (1 mark)

v. Draw the binary tree after deleting node 'reason'. (1 mark)

End of Page.